

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 06-161975

(43)Date of publication of application : 10.06.1994

(51)Int.Cl. G06F 15/16
G06F 11/20

(21)Application number : 04-306746

(71)Applicant : HITACHI LTD

(22)Date of filing : 17.11.1992

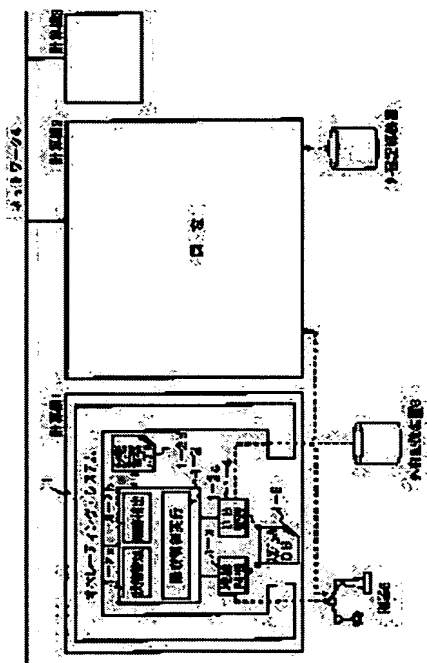
(72)Inventor : KOBAYASHI TAKASHI
MORI KINJI
MORI HIDEAKI
SUEKI MASAO

(54) COMPUTER SYSTEM

(57)Abstract:

PURPOSE: To eliminate the complication of configuration control processing, to decrease job interrupt time, and to improve the availability of a system by executing by deciding a processing schedule in which the optimum interrupt time according to the transfer of each job can be obtained.

CONSTITUTION: The execution notification of configuration control and state transition information are inputted from the configuration control execution means 1-2c of its own computer and another computer, and the states of its own computer and another computer are updated at every job. When the stoppage/restoration of the computer targeted to monitor is known, it is notified to the configuration control execution means 1-2c of its own or another computer. Thereby, a state is transferred to the one where the state of its own computer is designated by an instruction from an operator or another computer. Also, the state of its own computer is changed to a prescribed state based on the stoppage/restoration information of its own or another computer from a fault detecting means 1-2b. At this time, the feasibility of the configuration control is decided based on the state and every kind of execution constraint of each computer, and a schedule in which the optimum interrupt time according to the transfer of each job can be obtained is decided.



LEGAL STATUS

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

Copyright (C); 1998,2003 Japan Patent Office

BEST AVAILABLE COPY

(19)日本国特許庁(JP)

(12) 公開特許公報(A)

(11)特許出願公開番号

特開平6-161975

(43)公開日 平成6年(1994)6月10日

(51)Int.Cl.⁵

G 0 6 F 15/16
11/20

識別記号

4 7 0 B 9190-5L
3 1 0 F 7313-5B

庁内整理番号

F I

技術表示箇所

審査請求 未請求 請求項の数7(全 17 頁)

(21)出願番号 特願平4-306746

(22)出願日 平成4年(1992)11月17日

(71)出願人 000005108

株式会社日立製作所

東京都千代田区神田駿河台四丁目6番地

(72)発明者 小林 隆

神奈川県川崎市麻生区王禅寺1099番地 株
式会社日立製作所システム開発研究所内

(72)発明者 森 欣司

神奈川県川崎市麻生区王禅寺1099番地 株
式会社日立製作所システム開発研究所内

(72)発明者 森 英明

茨城県日立市大みか町五丁目2番1号 株
式会社日立製作所大みか工場内

(74)代理人 弁理士 小川 勝男

最終頁に続く

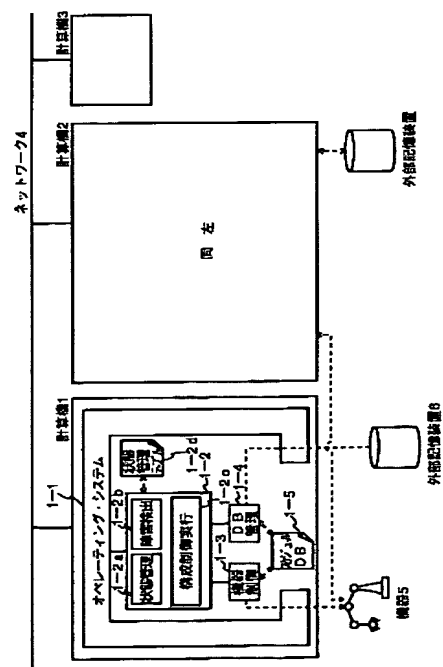
(54)【発明の名称】 計算機システム

(57)【要約】 (修正有)

【構成】複数の計算機により構成されたシステムにおいて、障害発生のため一方の計算機で実行していた機器制御業務とDB管理業務を他方の計算機に引き継ぐ際、後者の業務の引き継ぎ処理のために前者の業務が中断されるという問題を回避するために、各計算機に業務単位で構成制御を実行するための機能1-2を設ける。

【効果】構成制御処理の複雑性を低減し、システムの可用性を向上し構成制御中の業務中断時間の低減を図ることができる。

図 1



1

【特許請求の範囲】

【請求項1】複数の計算機により構成され、その要素である各計算機が複数の業務を実行するシステムであり、システムの一部の計算機が業務を実行できなくなった場合に他の計算機がその業務を引き継いで行うことができるシステムにおいて、前記各計算機の業務実行可否に関する内部状態を業務対応で管理し、各業務の引き継ぎ処理に伴う中断時間が最適となるような処理スケジュールを決定して実行することの特徴とする計算機システム。

【請求項2】請求項1において、前記システム内の前記各計算機が自計算機あるいは他計算機の内部状態を業務ごとに管理し、その変更情報または生死情報を入力して自計算機あるいは他計算機がとるべき内部状態を業務ごとに決定して、必要に応じて他計算機に内部状態を指示する計算機システム。

【請求項3】請求項1において、予め定められた一部の計算機がシステム内の各計算機の内部状態を業務ごとに管理し、その変更情報または生死情報を入力して各計算機がとるべき内部状態を業務ごとに決定して指示する計算機システム。

【請求項4】請求項1において、前記計算機が決定した内部状態、前記計算機以外の計算機から指示された内部状態、あるいは、オペレータから指示された内部状態に従って、前記計算機の内部状態を業務ごとに遷移する計算機システム。

【請求項5】請求項2において、業務単位でその業務が実行できるか否かを定期的に監視し、この情報に基づき、各計算機がとるべき内部状態を業務ごとに決定し、自計算機の内部状態を遷移するとともに、必要に応じて他計算機の状態遷移を指示する計算機システム。

【請求項6】請求項1において、システム内の前記計算機の一部に障害が発生した場合、その障害の種別に応じて各業務が実行できるか否かを判定し、その結果に基づいて業務単位にその引き継ぎを行う計算機システム。

【請求項7】請求項1において、各業務の緊急性とその引き継ぎに必要な処理の半順序性を制約として、各業務の引き継ぎに伴う中断時間が最適となるような処理スケジュールを決定して実行する計算機システム。

【発明の詳細な説明】

【0001】

【産業上の利用分野】本発明は複数計算機により構成するシステムにおいて、システムの一部が停止する場合の業務引き継ぎ方法に係り、特に、各計算機が複数の業務を兼務し、業務ごとに緊急性、信頼性、引き継ぎ方法などが異なる場合に、それらの特性に応じたタイミングで各業務を個別に引き継ぐ方法に関する。

【0002】

【従来の技術】近年、計算機の高性能、低価格化とネットワークの発達により、計算機制御システムの分散化が進んでいる。従来の分散制御システムは、機器の制御、

2

数値処理、スケジュール管理などシステム内の複数の計算機が各々単一の業務を実行する機能分散的なものであった。ところが、近年、ユーザのニーズが多様化して実行すべき業務の種類が増加したこと、また、計算機の処理能力が飛躍的に向上したことにより、システム内の各計算機で複数の業務を混在して実行することが通常の形態となってきた。

【0003】一方、制御システムの大規模化に伴い、システムのフォールトトレラント性へのニーズがますます高まっている。そのニーズにこたえるために、業務を実行するのに必要な処理プログラムあるいはデータを複数の計算機に多重にもつ、という情報の複写方式がとられる。この方式の実現方法は次のように業務によって異なる。

【0004】リアルタイム機器制御の業務では、対象となる機器類が停止しないように、制御要求が発生した時点から決められた時間内に制御情報を出力することが課題となる。そこで、同一の制御プログラム群を異なる複数台の計算機で並行して稼働させ、その結果出力される複数の制御情報のなかの一つを採用するという方法をとる。一部の計算機が停止しても他の正常な計算機が制御情報を出力するため業務を中断しなくてよい。

【0005】オンラインデータベース管理の業務では、データベースの一貫性とサービスの持続性の確保が課題となる。そこで、データベースのコピーを複数個作成し、それらを複数台の計算機（待機系とよぶ）に保管するという方法をとる。そして、マスタファイルの管理計算機（運転系とよぶ）が、ファイルの更新結果をある決められたタイミングで待機系に送出し、マスタファイルとそのコピーを一致させる。ディスク、あるいは、計算機の障害のためにマスタファイルがアクセスできなくなった場合、その時点までに運転系が受け付けた更新トランザクションをコピーファイルに反映させた後、待機系のうちの一つを新しい運転系に設定して処理を再開する。この方法では、待機系から運転系へ移行する際、ファイルの一貫性の保証、待機モードから運転モードへのプログラムの切替えなどを行う。このため、その間、業務を中断しなければならない。

【0006】構成制御とは、上記のように処理プログラム、データを多重化するために用いた計算機群の状態を管理し、システムの一部の計算機が障害、保守などのために停止した場合に、停止した計算機の切り離し、待機系計算機の運転系への移行などを行うものである。計算機状態として何を設定すべきかはそこで実行される業務に応じて決まる。例えば、リアルタイム制御業務では、多重化された各計算機が稼働中か停止中かを状態として設定すればよい。これにより、停止した計算機をシステムから切り離すことができる。一方、オンラインデータベース業務では、運転系と待機系ではその処理内容が異なるため、運転系で稼働中、待機系で稼働中、停止中と

3

いう三つの状態を設定する必要がある。

【0007】

【発明が解決しようとする課題】システム内の各計算機が単一の業務を行っていた場合には、その業務に応じて計算機状態を設定し、それに基づいて構成制御を行えばよかった。ところが、前述のように一つの計算機内で複数の業務を行うシステムでは、全ての業務に対応できる計算機状態を設定する必要がある。

【0008】最も単純な方法は、全業務の全状態を組み合わせて取り得る全ての状態を設定するものである。例えば、システム内の各計算機がリアルタイム制御業務とオンラインデータベース業務を兼務する場合、それぞれの計算機に例えば次のような状態を設定する必要がある。

【0009】

- 制御業務が稼動中でDB業務が運転系で稼動中
- 制御業務が稼動中でDB業務が待機系で稼動中
- 制御業務が稼動中でDB業務が待機系から運転系に移行中
- 制御業務が稼動中でDB業務が停止中
- 制御業務が停止中でDB業務が運転系で稼動中
- 制御業務が停止中でDB業務が待機系で稼動中
- 制御業務が停止中でDB業務が停止中

この方法では、計算機状態は兼務する業務の増加に伴って組み合わせ的に増加する。また、構成制御では、各計算機が自分の状態とその他の計算機の状態との組み合わせで実行すべき処理を決める。このため、構成制御で考慮すべき状態数は、単独の計算機に設定した状態数を計算機の数だけ累乗した数となる。このように状態数が増加すると、構成制御処理が複雑さが増えるものとなり、処理プログラムの信頼性や保守性が極めて悪くなる。

【0010】上記の複雑性の問題に対する解決策として、実行すべき全業務をできるのかできないのかという観点から、次のような縮退した計算機状態を設定する方法が考えられる。

【0011】

- 制御業務が稼動中でDB業務が運転系で稼動中
 - 制御業務が稼動中でDB業務が待機系で稼動中
 - 制御業務が停止中あるいはDB業務が停止中
- しかし、このように縮退した計算機状態を採用すると、制御業務とDB業務のうちのいずれか一方が実行できず、他の業務だけ実行するという場合に縮退運転ができない。また、運転系計算機に障害が発生した場合、待機系計算機が管理するバックアップDBの一貫性を保証して新たな運転系に移行する間、制御業務を中断しなければならない。

【0012】本発明の目的は、以上に述べたような問題を解決し、

- (1) 構成制御処理の複雑性の低減
- (2) システムの可用性の向上

4

(3) 構成制御中の業務中断時間の低減を実現することである。

【0013】

【課題を解決するための手段】上記の課題を解決するために、本発明においては業務実行可否に関する計算機状態を全ての業務に対応するように一元的に設定するのではなく、業務ごとに多元的に設定する。そして、各業務の緊急性とその引き継ぎに必要な処理の半順序性を制約として、各業務の引き継ぎに伴う中断時間が最適となるような処理スケジュールを決定して実行する。

【0014】そのために、次の手段を設定する。

(1) 計算機状態管理手段

計算機状態を業務単位に設ける。次に示す構成制御実行手段の実行通知を入力し、自計算機と他計算機の各業務の状態を管理する。

(2) 障害検出手段

他計算機あるいは自計算機の生死状態を監視する。必要ならばその結果を他計算機に通知する。

(3) 構成制御実行手段

上記の計算機状態管理手段で管理する計算機状態に基づき、他の計算機からの指示により、あるいは、自動的に構成制御を実行する。この際、各業務の緊急性とその引き継ぎに必要な処理の半順序性を制約として、各業務の引き継ぎに伴う中断時間が最適となるような処理スケジュールを決定する。

【0015】

【作用】上記の課題を解決するために次の方法をとる。

(1) 計算機状態管理

自計算機および他計算機の構成制御実行手段から、構成制御の実行通知と、その結果、計算機がどの状態からどの状態へ移行するかという状態遷移情報を入力する。これらの情報に基づき、自計算機および他計算機の状態を業務ごとに更新する。

(2) 障害検出

自計算機あるいは他計算機の生死監視を定期的に行う。これにより、監視対象とする計算機が停止／回復したことが判明した場合、自計算機あるいは他計算機の構成制御実行手段にこれを通知する。

(3) 構成制御実行

オペレータ、他計算機からの指示により、自計算機の状態を指定された状態に移行する。また、障害検出手段からの自計算機あるいは他計算機の停止／回復情報に基づき、自計算機の状態を所定の状態に自動的に変更する。この際、自計算機および他計算機の状態と各種実行制約に基づいて、構成制御の実行可否を判定する。また、各業務の引き継ぎに伴う中断時間が最適となるような処理スケジュールを決定する。

【0016】

【実施例】以下、本発明の実施例を図面に基づいて詳細に説明する。なお、以下に示す実施例では、二重系計算

機の各々が機器制御とDB管理という二つの業務を実行するシステムに適用した例を示す。

【0017】図1は、本発明の一実施例を示す二重系計算機システムの全体構成を示すブロック図である。図において、1は二重系のうちの一方の計算機であり、2は他方の計算機である。3は計算機1と計算機2が有するDBを更新するための小型計算機である。これら三つの計算機はネットワーク4で接続されている。二重系計算機1と2は全く同一の構成をとるものとする。以下、計算機1についてその内部構造を説明する。1-1はオペレーティングシステムであり、二台の計算機間の通信管理機能、外部記憶装置とのデータ入出力管理機能、および、制御すべき機器との信号入出力管理機能を含む。1-2は本発明の構成制御機能であり、計算機状態管理機能1-2a、障害検出機能1-2b、構成制御実行機能1-2c、計算機状態管理テーブル1-2dにより構成する。1-3は機器制御業務を実行する機能であり、1-4はDB管理業務を実行する機能である。1-5は機器制御を行うためのスケジュールDBであり、DB管理業務が二重系計算機以外の計算機からのトランザクションに応じて時々刻々と更新し、制御業務が随時これを参照する。5は制御機能1-3が制御すべき機器であり、6はDB1-5のバックアップのための外部記憶装置である。機器5は二重系計算機の両方と接続されており、いずれかの制御信号を選択的に入力するものとする。

【0018】二重系計算機は機器制御とスケジュールDB管理という二つの業務を行う。まず、機器制御業務について説明する。制御すべき機器は、多品種の部品の中からスケジュールで定められた部品を選択し、その組立作業を行うものである。計算機1の制御機能1-3では、スケジュールDB1-5を参照して次に組み立てるべき部品を把握し、機器5にその情報を送出して組立作業の開始を指示する。計算機2も同様な方法で機器5に対して組立作業を指示する。機器5は、計算機1および計算機2から送出された制御出力のうち一方（本例では計算機1の制御出力）を採用し、それに基づいて組立作業を実行する。そして、終了信号を制御機能1-3および計算機2の制御機能に対して送出する。

【0019】次に、DB管理業務について説明する。本業務は、スケジュールDBのオンライン更新の管理と外部記憶装置へのバックアップ取得を行うものである。本例では、計算機1を運転系、計算機2を待機系と設定することにする。DB管理機能1-4では、計算機3から送信されたスケジュールの訂正、追加、削除などの要求を受け付け、それに応じてスケジュールDB1-5を更新する。また、予め定められたタイミングで、その更新結果を外部記憶装置6、および、計算機2の管理するバックアップDB（メモリ上のものと外部記憶装置上のもの）に反映させる。

【0020】二重系計算機のうちの一方が停止した場合

の、各業務の引き継ぎ方法を述べる。ここでは、計算機1が停止した場合を想定する。制御業務では、計算機1と計算機2が全く同じ処理を行っているため、業務の引き継ぎは次のように一瞬にして行うことができる。すなわち、機器5に対して、計算機1の制御出力を採用するのでなく、計算機2の制御出力を採用するように指示すればよい。

【0021】一方、DB管理業務では、計算機1がメインファイル1-5を管理し、計算機2がそのバックアップファイル进行管理しているため、業務の引き継ぎには次の一連の処理が必要である。

【0022】（1）計算機3に対して、運転系が計算機1から計算機2に変わったことを通知する。

【0023】（2）計算機1がDB1-5を更新した結果を、計算機2のDB2-5に反映させる。

【0024】（3）計算機2のDB管理機能のうち、バックアップファイル管理機能を停止し、メインファイル管理機能を起動する。

【0025】以上のような業務引き継ぎ制御を行うためには、二重系の各計算機がどのような内部状態にあるかを把握し、それに応じた処理を行う必要がある。図2は、各業務ごとに設定した状態とそれらの遷移関係を示したものである。制御業務の状態は、「稼動」と「停止」の二つであり、DB管理業務の状態は、「運転」、「待機」、「停止」の三つである。

【0026】図3は、二重系の複合的な状態遷移関係を示したものである。図において、ノードは二重系の複合状態を示し、左半分は計算機1の状態を、右半分は計算機2の状態を表す。アークは状態間の遷移を示し、その横に記された文字は状態遷移を開始するイベントを示す。業務の引き継ぎが必要な遷移は太線で示してある。

【0027】以上に述べたシステム構成、業務内容、および、業務の引き継ぎ方法を前提として、構成制御を業務ごとに実行する方法を具体的に説明する。

【0028】図4に従って構成制御の実行手順を説明する。本処理は、オペレータ指示、障害検出などのイベントにより起動される。業務ごとに起動イベントを変えることにより、構成制御の実行タイミングを各業務にとって最適なものにすることができる。

【0029】ステップ401：構成制御の実行指示、発生したイベント名称を受け付け、図3をもとに、状態遷移を実行すべき業務の名称と遷移先の状態名称を決定する。

【0030】ステップ402：計算機状態テーブルから各業務の現在の状態を入力する。

【0031】ステップ403：ステップ401とステップ402の入力情報に基づいて、予め定めた実行制約を満足する可否かを判定する。図5に実行制約の例を示す。これは、図3の状態遷移図を表形式に書き直したものである。表の横軸は遷移前の状態であり、縦軸は遷移

後の状態である。○印のない遷移は実行することができないことを示す。実行制約を満足する場合はステップ404に進み、そうでない場合は処理を終了する。

【0032】ステップ404：構成制御を実行する前に、計算機状態テーブルに状態遷移中であることを登録する。図6に計算機状態テーブルを示す。本テーブルは計算機1と計算機2に各々設定し、各計算機が自計算機と他計算機の状態を業務ごとに管理する。テーブルに登録する値は、図3で示した各状態と、構成制御実行中であることを示す「遷移中状態」である。

【0033】ステップ405：ステップ401で入力した情報に対応した構成制御を実行する。例えば、「運転系停止イベント」が発生した場合、前述の様に次の処理を実行する。

【0034】(1) 計算機3に対して、運転系が計算機1から計算機2に変わったことを通知する。

【0035】(2) 計算機1がDB1-5を更新した結果を、計算機2のDB2-5に反映させる。

【0036】(3) 計算機2のDB管理機能のうち、バックアップファイル管理機能を停止し、メインファイル管理機能を起動する。

【0037】ステップ406：構成制御の終了後、計算機状態テーブルに新たな状態に遷移したことを登録する。以上で処理を終了する。

【0038】図7に従って、計算機状態管理手順を示す。本処理は、上述の構成制御実行処理のステップ404と406において、計算機状態テーブルに状態を参照および登録する際にコールする。参照および登録処理が多重に実行されることを防止するために、処理前にロックを設定し、処理後にそのロックを解除する。

【0039】ステップ701：業務の名称、遷移先の状態名称を引数として受け付ける。この状態名称は、ステップ404でコールされた場合は「遷移中」であり、ステップ406でコールされた場合は新たに遷移する状態の名称である。

【0040】ステップ702：計算機状態テーブルの、状態を登録すべき計算機の業務の欄にロックを設定する。

【0041】ステップ703：計算機状態テーブルの該当する欄に遷移先の状態名称を登録する。

【0042】ステップ704：ステップ702で設定したロックを解除する。

【0043】図8および図9に従って、障害検出手順を説明する。本例では、二重系の各計算機がお互いに相手に向けて定期的に信号を送出し、各計算機はこの信号を定期的に受信する。この信号を生死信号とよぶ。両方の計算機が稼動している場合は、お互いに相手の送出した生死信号をある一定時間内に受信できる。いずれか一方が停止した場合には、稼動計算機の受信処理はタイムアウトとなり、これにより相手計算機が停止したことを認

識できる。

【0044】まず、図8に従って送信側の処理手順を述べる。

【0045】ステップ801：計算機状態テーブルの内容を読み取る。

【0046】ステップ802：上記の情報を生死信号にのせて相手計算機に送出する。

【0047】ステップ803：予め定められた時間だけ待つ。その後、ステップ801に戻る。

10 【0048】次に、図9に従って受信側の処理手順を述べる。

【0049】ステップ901：受信用のバッファを参照し、生死信号を受信したか否かを判定する。受信した場合はステップ905に進み、そうでない場合はステップ902に進む。

【0050】ステップ902：タイマを参照し、その値が予め定めた値を越えている（タイムアウト）か否かを判定する。タイムアウトの場合はステップ908に進み、そうでない場合はステップ903に進む。

20 【0051】ステップ903：タイマの値を1単位分増加する。

【0052】ステップ904：予め定められた時間だけ待つ。その後、ステップ901に戻る。

【0053】ステップ905：タイマの値を初期値にリセットする。

【0054】ステップ906：バッファの内容をクリアする。

【0055】ステップ907：予め定められた時間だけ待つ。その後、ステップ901に戻る。

30 【0056】ステップ908：タイマの値を初期値にリセットする。

【0057】ステップ909：相手計算機が停止したと判定。

【0058】ステップ910：構成制御機能に対して相手計算機の停止を通知し、処理を終了する。

【0059】図10と図11に従って、構成制御処理の実行スケジュールの決定方法を説明する。図10は、引き継ぎ処理を行う際を守るべき半順序関係を、三つの異なる業務ごとに示したものである。○印が各処理を示し、矢印は処理間の順序性を規定する。例えば、業務1の引き継ぎでは、処理Aは処理Bの後、処理Dは処理Bと処理Cの後に実行する必要がある。処理Bと処理Cはどちらを先に実行してもよい。また、簡単のために業務間に優先度はないものとする。優先度のある場合の方法は後述する。このような処理順序関係をもとに、最適な処理スケジュールを決定する方法を以下に説明する。

【0060】ステップ1101：業務ごとに、引き継ぎ処理の順序関係をグラフにかく（図10(a)参照）。

50 【0061】ステップ1102：業務ごとの順序関係グラフを一つのグラフに合成する（図10(b)参照）。

【0062】ステップ1103：合成された半順序関係をもとに、引き継ぎ処理で取り得る処理系列を列挙する（図10（c）参照）。

【0063】ステップ1104：系列ごとに、各業務の中断時間を算出する。

【0064】ステップ1105：中断時間のトータルが最小の系列を最適なものとして選択する。本例では、CABDEという系列が最適となる（図10（c）参照）。

【0065】以上は業務に優先度がない場合の方法である。優先度がある場合は、ステップ1105において、優先度が高い業務ほど大きいウェイトを掛け合わせればよい。例えば、業務3の優先度が他の業務よりも高い場合、例えば、業務3には10のウェイトを、他の業務には1のウェイトを掛け合わせる。この場合、最適系列はAEB CDおよびCAEB Dとなる。

【0066】最後に、本発明の構成制御方式により従来方式に比べてどのような効果が得られるかを、図1から図9で述べたシステムを例として示す。

【0067】まず、処理の複雑性を低減する効果を示す。「発明が解決しようとする課題」の部分で述べたように、複数の業務を兼務する場合の計算機状態の設定方法には次の二つがある。

【0068】（1）全業務の全状態を組み合わせて取り得る全ての状態を設定する。

【0069】（2）実行すべき全業務をできるのかできないのかという観点から、縮退した計算機状態を設定する。

【0070】図2に示したように、制御業務の状態数は二つ、DB業務の状態数は三つである。（1）の方法をとった場合、両業務に対応する状態数は6（ 2×3 ）となり、二重系の複合状態数は36（ 6×6 ）となる。これに対して、本発明のように業務ごとに状態遷移を考えた場合、制御業務の複合状態数は4（ 2×2 ）、DB業務の複合状態数は9（ 3×3 ）となる。明らかに、36状態を考慮する複雑な構成制御処理を考えるよりは、4状態と9状態の比較的単純な処理を考える方が、設計上の誤りが少なく、保守性も向上する。また、（2）の方法をとった場合、前述したように、両業務に対応する状態数は三つとなり、複合状態数は9（ 3×3 ）となる。これは、本発明の方法を適用した場合の状態数と同程度の数であり、構成制御処理を比較的単純に実現できる。しかし、以下に述べるように、システムの可用性、構成制御中の業務中断時間に問題がある。

【0071】次に、構成制御中の業務中断時間に関する効果を述べる。図12と図13は、計算機1が障害で停止し、計算機2が業務の引き継ぎを行う際の処理と状態遷移のタイムチャートを示す。図12は上記の従来方法

（2）で状態を設定した場合のものであり、図13は本発明の方法で状態を設定した場合のものである。図12に示すように、従来方法では、全ての業務の引き継ぎが完了するまで新たな状態に遷移できない。このため、制御業務は、その引き継ぎ処理が短時間でできるにもかかわらず、DB管理業務の引き継ぎが完了するまで中断される。DBの一貫性を保証するための処理は、その方法によっては数秒から数十秒を要するものとなる。機器制御に要求される応答時間がそれよりも短い場合、機器が停止してしまい、二重系構成にした意味がなくなる。図13に示すように、本発明の方法によれば、業務単位で引き継ぎ処理を行うことができるため、制御業務がDB管理業務のために中断されることはなくなる。

【0072】最後に、システムの可用性に関する効果を述べる。二重系計算機的一方に障害が発生した場合、その障害の程度によっては一部の業務を停止して縮退運転を行うことがある。例えば、図1のシステムにおいて機器5と計算機2をつなぐラインが切断された場合、計算機2は制御業務だけ停止してDB管理業務を続行する。このような場合、従来方法（2）では対処できない。本発明の方法によれば、このような業務ごとの停止／回復は容易である。

【0073】

【発明の効果】本発明によれば構成制御処理の複雑性が低減し、システムの可用性が向上し、構成制御中の業務中断時間が低減する。

【図面の簡単な説明】

【図1】本発明を適用した二重系計算機システムのブロック図。

【図2】各計算機の単独の状態遷移図。

【図3】二重系計算機の複合状態遷移を示す説明図。

【図4】構成制御実行手順を示すフローチャート。

【図5】実行制約の例の説明図。

【図6】計算機状態管理テーブルを示す説明図。

【図7】計算機状態管理手順のフローチャート。

【図8】障害検出のための送信側の手順のフローチャート。

【図9】障害検出のための受信側の手順を示すフローチャート。

【図10】構成制御の最適処理順序の決定方法の説明図。

【図11】構成制御の最適処理順序の決定方法の説明図。

【図12】本発明の効果の説明図。

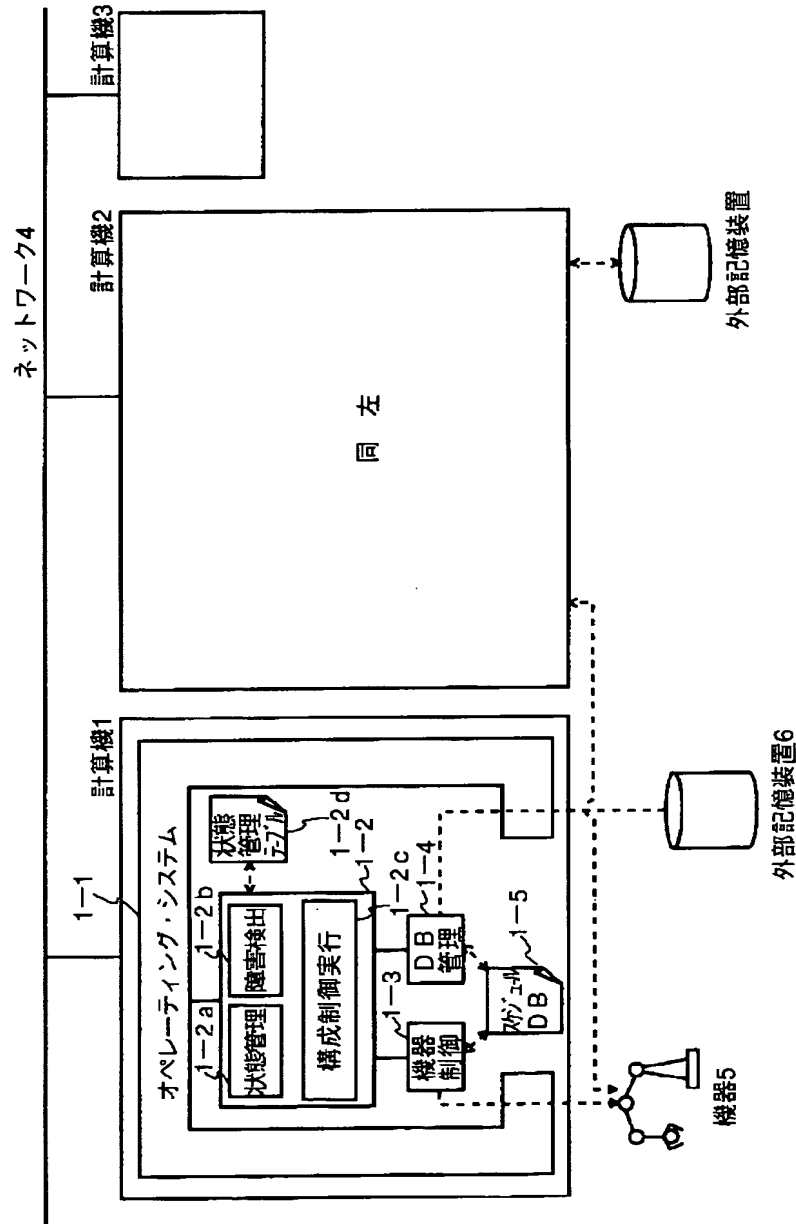
【図13】本発明の効果の説明図。

【符号の説明】

1, 2, 3…計算機、4…ネットワーク、5…機器、6…外部記憶装置。

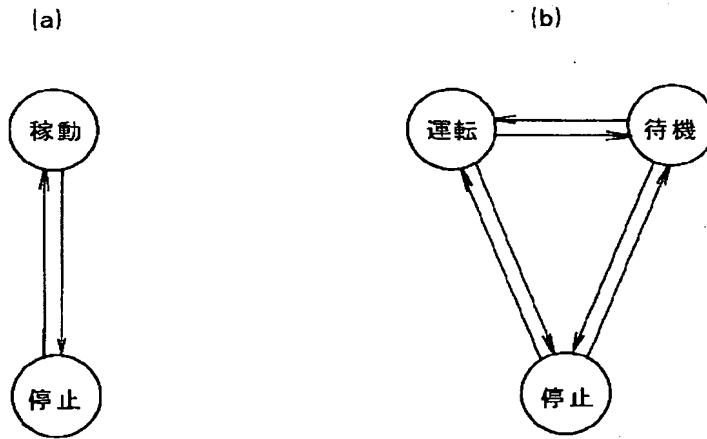
【図1】

図 1



【図2】

図 2

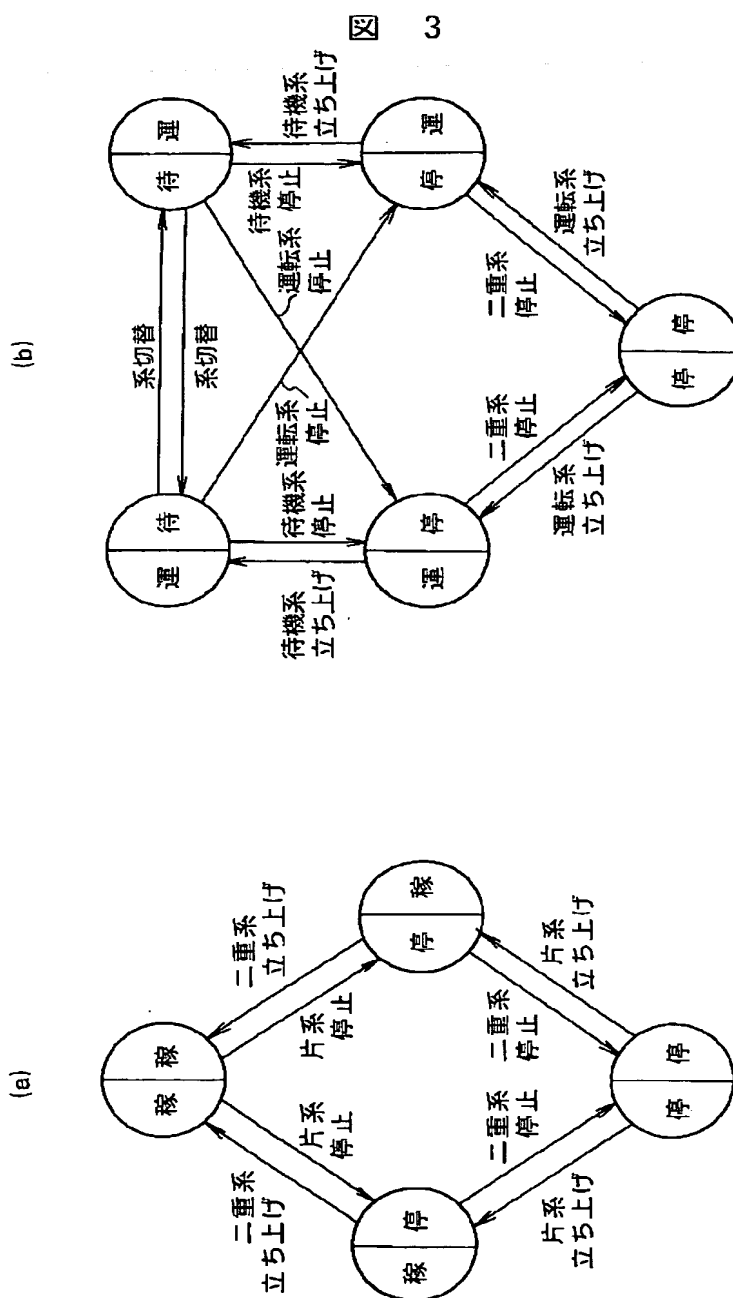


【図6】

図 6

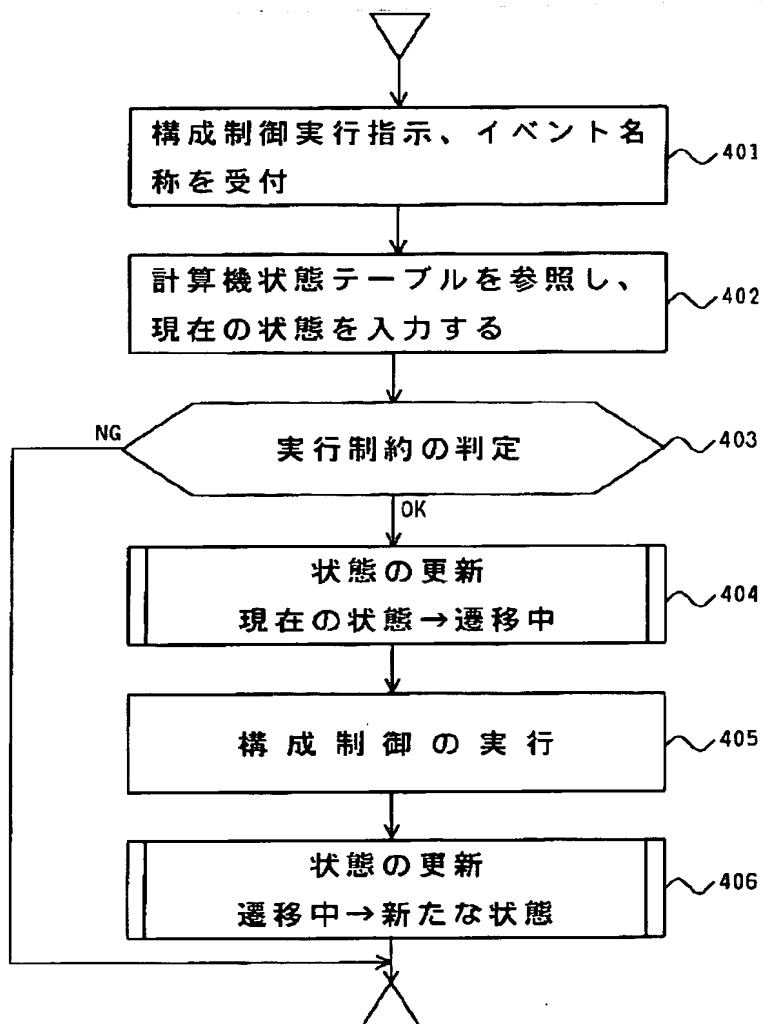
計算機 \ 業務	制 御 業 務		D B 管理業務	
		ロ ッ ク		ロ ッ ク
計算機 1	稼 動	OFF	運 転	OFF
計算機 2	遷 移 中	ON	待 機	OFF

【図 3】



【図4】

図 4



【図5】

図 5

(a)

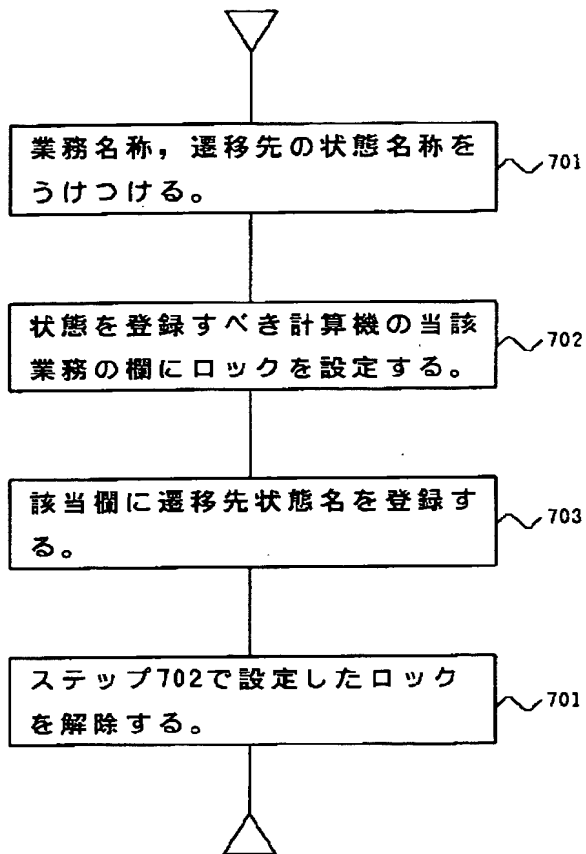
遷移後 遷移前	稼, 稼	稼, 停	停, 稼	停, 停
稼, 稼		○ 片系停止	○ 片系停止	
稼, 停	○ 二重系 立上げ			○ 二重系 停止
停, 稼	○ 二重系 立上げ			○ 二重系 停止
停, 停		○ 片系 立上げ	○ 片系 立上げ	

(b)

遷移後 遷移前	運, 待	待, 運	運, 停	停, 運	停, 停
運, 待		○ 系切替	○ 待機系 停止	○ 運転系 停止	
待, 運	○ 系切替		○ 運転系 停止	○ 待機系 停止	
運, 停	○ 待機系 立上げ				○ 二重系 停止
停, 運		○ 待機系 立上げ			○ 二重系 停止
停, 停			○ 運転系 立上げ	○ 運転系 立上げ	

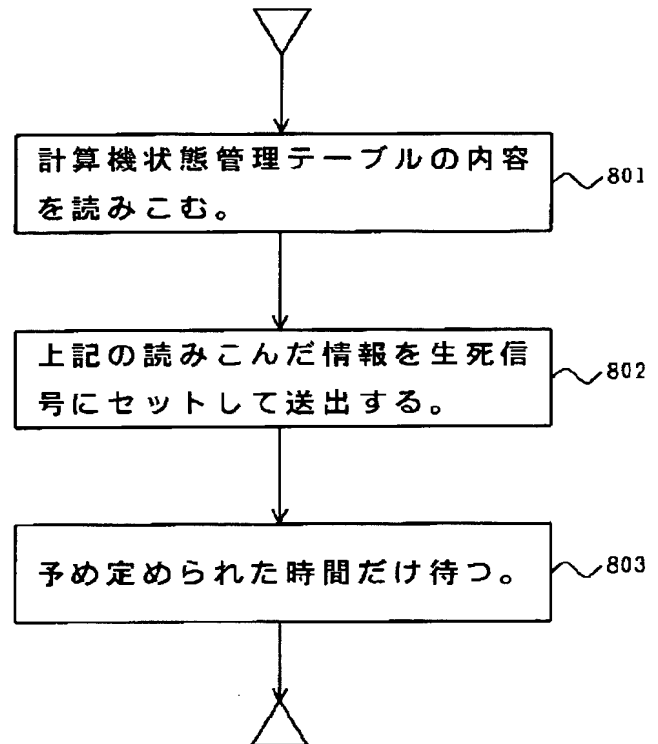
【図7】

図 7



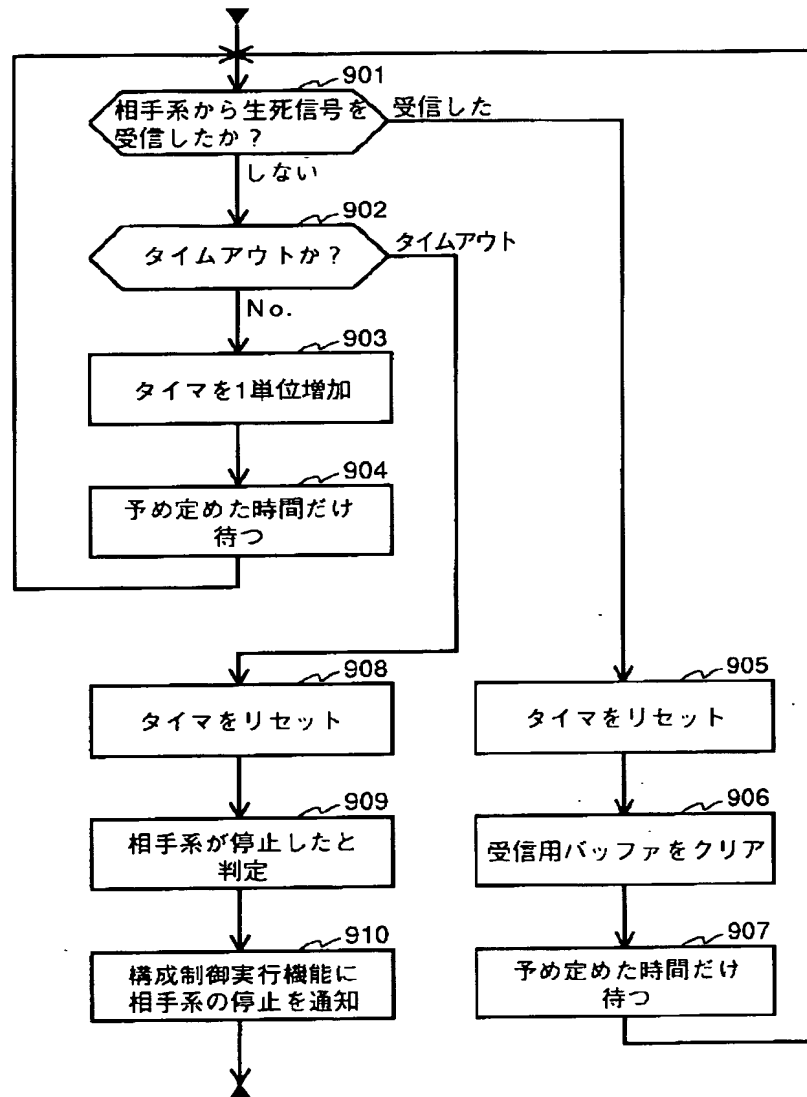
【図8】

図 8



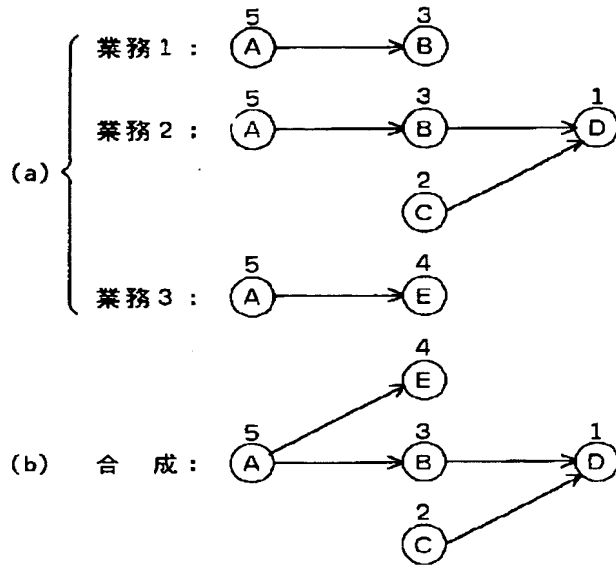
【図9】

図 9



【図10】

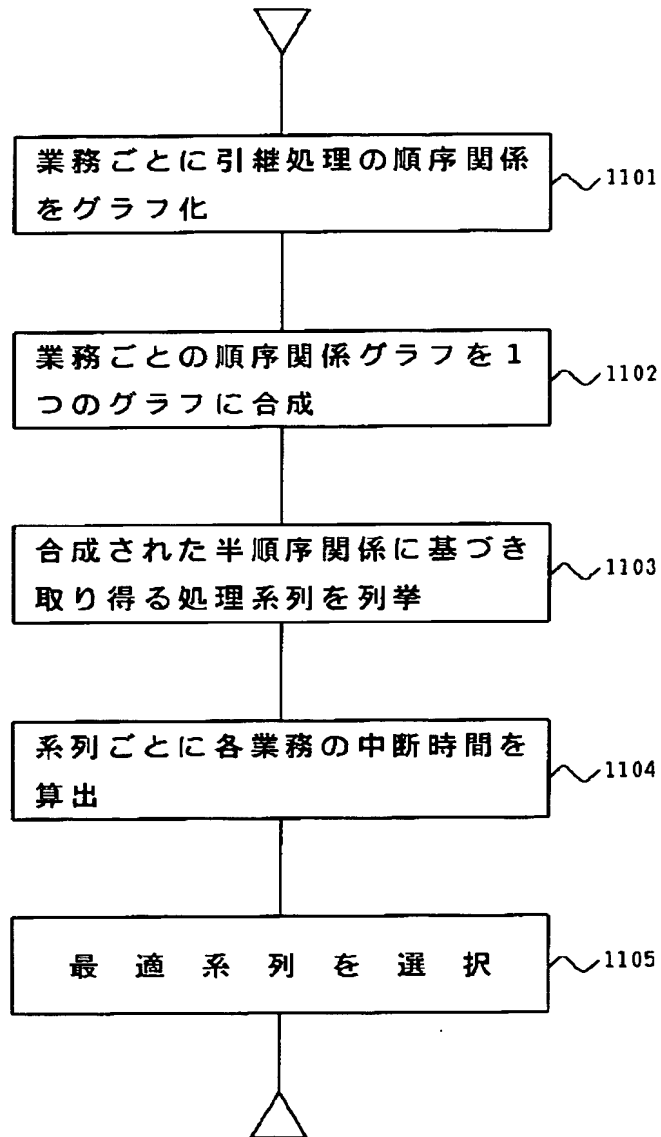
図 10



処理系列		中断時間			
		業務 1	業務 2	業務 3	トータル
(c)	A, B, C, D, E	8	11	15	34
	A, B, C, E, D	8	15	14	37
	A, B, E, C, D	8	15	12	35
	A, C, B, D, E	10	11	15	36
	A, C, B, E, D	10	15	14	39
	A, E, B, C, D	12	15	9	36
	A, E, C, B, D	14	15	9	38
	C, A, B, D, E	8	11	13	32
	C, A, B, E, D	8	15	12	35
	C, A, E, B, D	12	15	9	36

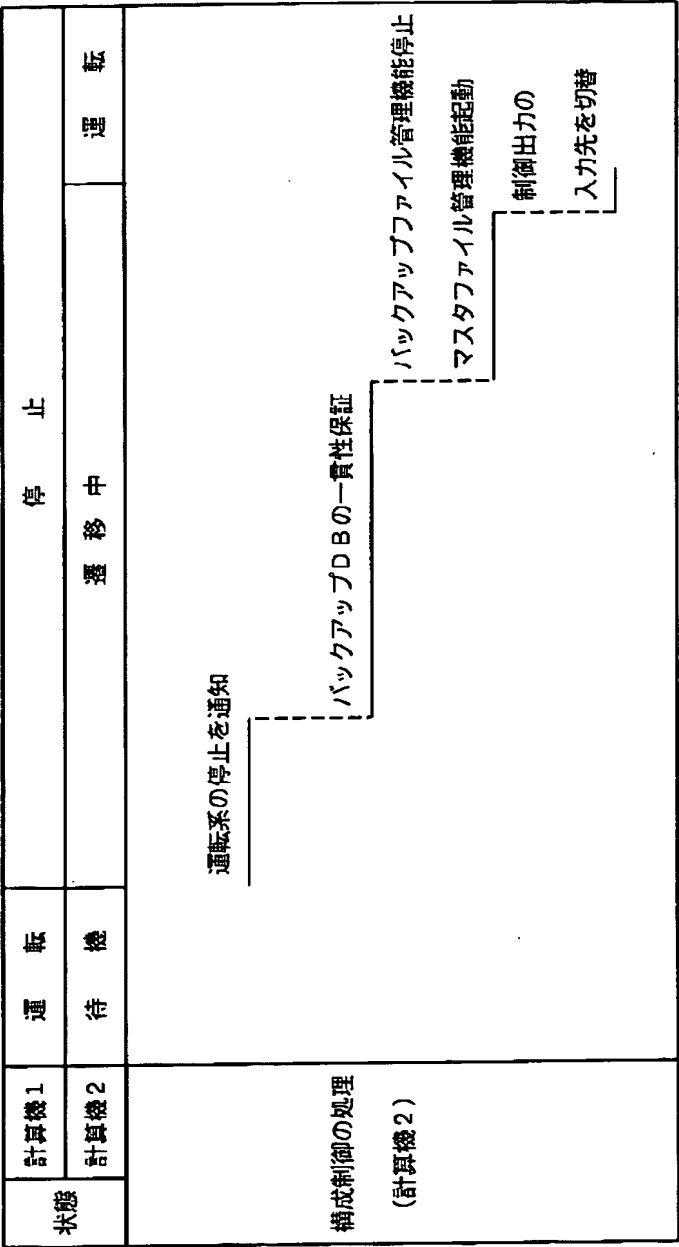
【図11】

図 11



【図12】

図 12



【図13】

図 13

